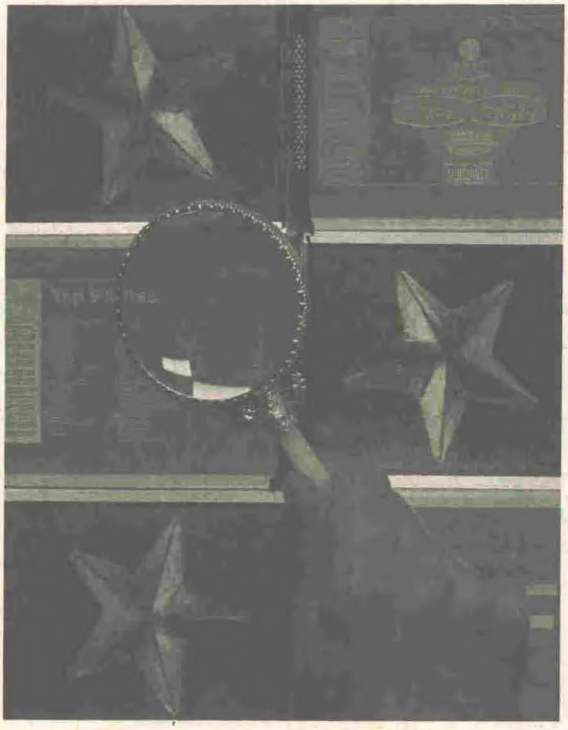
VISUAL BASIC

ขจร โรจนเมธินทร์

VB4 : จัดการดาต้าเบสด้วยวิชวลเบสิก 4



ใน VB4 มีสิ่งที่เปลี่ยนแปลงไปจาก VB3 ช่วยให้เราเขียนโปรแกรมได้ง่าย และมีระเบียบมากขึ้น

ปกติการจัดการแอ็กเซสดาต้าเบสนั้นทำได้ 3 วิธีคือ ใช้ไมโครซอฟต์แอ็กเซส ใช้แอพพลิเคชัน Data Manager ที่อยู่ในชุดวิชวลเบสิก และใช้คำสั่งของวิชวลเบสิกจัดการเอง สำหรับวิธีนี้ผู้เขียนเคยนำมากล่าวไปครั้งหนึ่งแล้ว แต่ในวิชวลเบสิก 4 ก็มีสิ่งที่เปลี่ยนแปลงไปจากวิชวลเบสิก 3 บ้าง โดยเฉพาะสิ่งที่เปลี่ยนนี้ช่วยให้เราเขียนโปรแกรมได้ง่ายและมีระเบียบมากขึ้น แม้ว่าขณะนี้มีวิชวลเบสิก 5 เริ่มเข้ามาขายในเมืองไทยแล้ว แต่ผู้เขียนคิดว่าไมโครซอฟต์คงยังไม่รีบเปลี่ยนแปลงแนวคิดเกี่ยวกับดาต้าเบสให้ต่างไปจากเดิมมากนัก จึงน่าจะเป็นประโยชน์หากเราจะทำความเข้าใจการจัดการดาต้าเบสด้วยวิชวลเบสิกอีกครั้ง

มีอะไรใหม่ในวิชวลเบสิก 4

ความจริงถ้าจะใช้วิธีการสร้างดาต้าเบสเดิมใน VB3 มาใช้กับ VB4 ก็ยังทำได้ แต่ถ้าจะเขียนในสไตล์ของ VB4 จริง ๆ ก็จะเห็นว่ามีการเปลี่ยนแปลงของเมธอดที่ใช้สร้างดาต้าเบสไปมากพอสมควร โดยพอจะสรุปได้ดังนี้

- มีเมธอด CreateTableDef และ CreateField สำหรับกำหนดเทเบิลและฟิลด์
- สามารถกำหนดการตรวจสอบข้อมูล (validation) ในระดับฟิลด์และเทเบิลได้ด้วยการกำหนดพรอพเพอร์ตี้สำบางค่าให้เหมาะสม
- สามารถสร้างรีเลชันระหว่างเทเบิลและบังคับให้มี referential integrity
- การใช้ประโยค SQL เพื่อกำหนดดาต้านั้นสามารถสร้างและแก้ไขเทเบิลได้
- ผู้ใช้สามารถลบฟิลด์จากเทเบิลภายใต้เงื่อนไขที่กำหนดได้

เขียนโปรแกรมเพื่อสร้างดาต้าเบส

การสร้างดาต้าเบสขึ้นใช้ภายในแอพพลิเคชันนั้นมีประโยชน์มาก เพราะช่วยให้ไม่ต้องกังวลในการติดตั้งดาต้าเบสไฟล์ให้ถูกต้อง เพราะตัวแอพพลิเคชันเองเป็นคนสร้างดาต้าเบสขึ้นเองขณะที่ทำงาน ขั้นตอนการสร้างดาต้าเบสมีดังนี้

1. ใช้ประโยค Dim กำหนดดาต้าเบสออบเจ็กต์ขึ้นมาใหม่
2. ใช้เมธอด CreateDatabase เพื่อสร้างดาต้าเบส
3. ใช้ประโยค Dim กำหนดออบเจ็กต์ TableDef และเมธอด CreateTableDef สร้างเทเบิล
4. กำหนดพรอพเพอร์ตี้สำของเทเบิลใหม่ที่สร้าง
5. สร้างฟิลด์และอินเด็กซ์ออบเจ็กต์ด้วยประโยค Dim และเมธอด CreateField และ CreateIndex

6. กำหนดพรอพเพอร์ตี้ส์ของฟิลด์และอินเด็กซ์
7. ใช้เมธอด Append ใส่ฟิลด์และอินเด็กซ์เพิ่มไว้ในเทเบิล
8. ใช้เมธอด Append ใส่เทเบิลไว้ในดาต้าเบส

การสร้างดาต้าเบสใหม่

วิธีเขียนโปรแกรมสร้างดาต้าเบสในตอนนี้ ผู้เขียนขอยกตัวอย่างการสร้างดาต้าเบสตามโครงสร้างของเทเบิลที่แสดงในตารางที่ 1

ฟิลด์	ชนิด	ความยาว
PersonID	Numeric	Integer
FirstName	Character	30
LastName	Character	30
Address	Character	50
Department	Character	30
Position	Character	30
Salary	Numeric	Integer

ตารางที่ 1 แสดงโครงสร้างเทเบิลตัวอย่างที่สร้าง

เดิมการสร้างดาต้าเบสใหม่ด้วย VB3 จะใช้ประโยค Create Database แต่ใน VB4 นั้นมีออบเจ็กต์ DBEngine และ Workspace เพิ่มมาใหม่สำหรับการเรียกใช้ดาต้าเบส นอกจากนี้ประโยค CreateDatabase ก็เปลี่ยนไปเป็นเมธอดของออบเจ็กต์ Workspace แทน ปกติแล้วทุกครั้งที่เราใช้แจ็ทดาต้าเบสเอ็นจินก็จะมีการสร้างออบเจ็กต์ Workspace ให้โดยอัตโนมัติ และถ้าจำเป็นผู้ใช้อาจสร้างเพิ่มเติมได้ โดยที่แต่ละเวอร์กสเปซจะมีหมายเลขประจำตัวสำหรับอ้างถึง

```
Dim NewDb As Database, NewWs As Workspace
Set NewWs = DBEngine.Workspaces(0)
Set NewDb =
NewWs.CreateDatabase("A:\PERSON.MDB",dbLangGeneral)
```

ส่วนโปรแกรมที่แสดงข้างบนนี้ทำหน้าที่สร้างดาต้าเบสใหม่ เริ่มจากกำหนดเวอร์กสเปซใหม่คือ NewWs ขึ้นมาก่อน จากนั้นจึงใช้เมธอด CreateDatabase สร้างดาต้าเบสชื่อ PERSON.MDB ที่สำคัญต้องไม่ลืมส่งค่า dbLangGeneral ให้เมธอดเพื่อกำหนดภาษาและโค้ดเพจของดาต้าเบสที่สร้าง นอกจากนี้ยังมีอาร์กิวเมนต์ที่ใช้ระบุว่าจะสร้างแอ็กเซสดาต้าเบสรุ่น 1.0, 1.1, 2.0 หรือ 3.0 ก็ได้ และอาร์กิวเมนต์ที่กำนดการเอ็นคริพต์ข้อมูลในดาต้าเบสด้วย วิธีกำหนดทางเลือกทั้งสองอันนี้ทำโดยส่งค่าที่ได้จากการบวกกันของทางเลือกแล้วส่งให้กับอาร์กิวเมนต์ตัวสุดท้ายของเมธอด แต่มีเงื่อนไขอาร์กิวเมนต์ต้องเป็น Long Integer ตัวอย่างเช่น ต้องการสร้างดาต้าเบสรุ่น 1.1 และทำการเข้ารหัสก็ต้องเขียนโค้ดดังนี้

```
Dim NewDb As Database, NewWs As Workspace
Set NewWs = DBEngine.Workspaces(0)
Dim DbOpts As Long
DbOpts = dbVersion11 + dbEncrypt
Set NewDb = NewWs.CreateDatabase("A:\PERSON.MDB",
dbLangGeneral, DbOpts)
```

ข้อควรระวังในการใช้เมธอด CreateDatabase ก็คือ ถ้าชื่อไฟล์ที่เราต้องการสร้างมีอยู่ก่อนแล้ว จะทำให้เกิดเออเรอร์ขึ้นได้ ดังนั้นจึงต้องเขียนโค้ดเพื่อจัดการกับเออเรอร์ที่อาจเกิดขึ้นไว้เสมอ หรืออาจใช้วิธีตรวจสอบชื่อไฟล์ก่อนที่จะสร้างดาต้าเบสใหม่ก็ได้

การสร้างเทเบิลใหม่

ในขั้นตอนการสร้างดาต้าเบสนั้น เรายังไม่สามารถใช้งานได้จนกว่าจะสร้างเทเบิลและใส่เทเบิลนั้นไว้ในดาต้าเบส ขั้นแรกของการสร้างเทเบิลก็คือ กำหนดออบเจ็กต์ TableDef ขึ้นก่อน เราจะกำหนดพรอพเพอร์ตี้ส์เทเบิลที่จะสร้างกับออบเจ็กต์ TableDef เดิมใน VB3 จะต้องใช้คีย์เวิร์ด New ร่วมกับประโยค Set เสมอ แต่ใน VB4 ไม่จำเป็น เราอาจเขียนโค้ดดังนี้

```
Dim NewTbl As TableDef
Set NewTbl = NewDb.CreateTableDef("Persons")
```

ผู้อ่านจะเห็นว่าในโค้ดข้างบนนี้เราต้องกำหนดพรอพเพอร์ตี้ส์ Name ของเทเบิลที่สร้างเสมอ แต่กรณีที่จะ Attach เทเบิลจากภายนอกมาไว้ในดาต้าเบสก็อาจต้องกำหนดค่าพรอพเพอร์ตี้ส์ Attribute, Connect, SourceTableName ด้วย หลังจากสร้างเทเบิลเรียบร้อยแล้ว ผู้ใช้อาจกำหนดพรอพเพอร์ตี้ส์อื่น ๆ ในภายหลังได้ เช่น กำหนดเงื่อนไขสำหรับตรวจสอบข้อมูล และข้อความเตือน เช่น

```
NewTbl.ValidationRule = "Salary < 20000 "
NewTbl.ValidationText = "Salary must not exceed
20000 baht."
```

การกำหนดฟิลด์

หลังจากกำหนดออบเจ็กต์ TableDef แล้ว ต่อไปจะต้องกำหนดออบเจ็กต์ Field โดยที่ในแต่ละเทเบิลจะบรรจุฟิลด์ได้มากกว่าหนึ่งฟิลด์ ฟิลด์แต่ละอันต้องกำหนดชื่อและชนิดไว้เสมอ สำหรับฟิลด์บางชนิดก็อาจต้องกำหนดพรอพเพอร์ตี้ส์อื่นเพิ่มเติมด้วย เช่น ฟิลด์ที่เป็นเท็กซ์จะต้องกำหนดความยาวไว้ ความยาวที่เป็นได้อยู่ในช่วง 1 ถึง 255 หากต้องการเก็บเท็กซ์ที่ยาวกว่านี้ต้องใช้เป็นชนิด Memo แทน สำหรับการกำหนดพรอพเพอร์ตี้ส์ Attributes นั้น หากเลือกแบบ Auto-increment (dbAutoIncrField) ทุกครั้งที่มีการเพิ่มเรคอร์ดใหม่ค่าในฟิลด์นั้นจะถูกเพิ่มขึ้นตาม ปกติเราใช้วิธีการนี้เป็นตัวนับเรคอร์ด นอกจากนี้ยังอาจใช้เป็น Primary key ได้เพราะมีค่าไม่ซ้ำกันแน่นอน ฟิลด์ที่จะใช้เป็นตัวนับนี้ต้องเป็นข้อมูลชนิด Long เท่านั้น ต่อไปนี้เป็นโค้ดที่จะสร้างเทเบิลตามที่กำหนดในตารางที่ 1

```
Dim Fd1 As Field, Fd2 As Field, Fd3 As Field,
Fd4 As Field
Dim Fd5 As Field, Fd6 As Field, Fd7 As Field
'*********************************************************
'Specify field name, type, and size as
CreateField arguments
'*********************************************************
Set Fd1 = NewTbl.CreateField("PersonID",
dbInteger)
Set Fd2 = NewTbl.CreateField("FirstName",
dbText, 30)
Set Fd3 = NewTbl.CreateField()
'*******************************
'Explicitly set field properties
'*******************************
Fd3.Name = "LastName"
Fd3.Type = dbText
Fd3.Size = 30
Set Fd4 = NewTbl.CreateField("Address",
dbText, 50)
Set Fd5 = NewTbl.CreateField("Department",
dbText, 30)
Set Fd6 = NewTbl.CreateField("Positon",
dbText, 30)
'************************************
'Set validation properties for a field
'************************************
Set Fd7 = NewTbl.CreateField()
Fd7.Name = "Salary"
Fd7.Type = dbInteger
Fd7.ValidationRule = "Salary < 20000"
Fd7.ValidationText = "Salary must not be
greater than 20000."
```

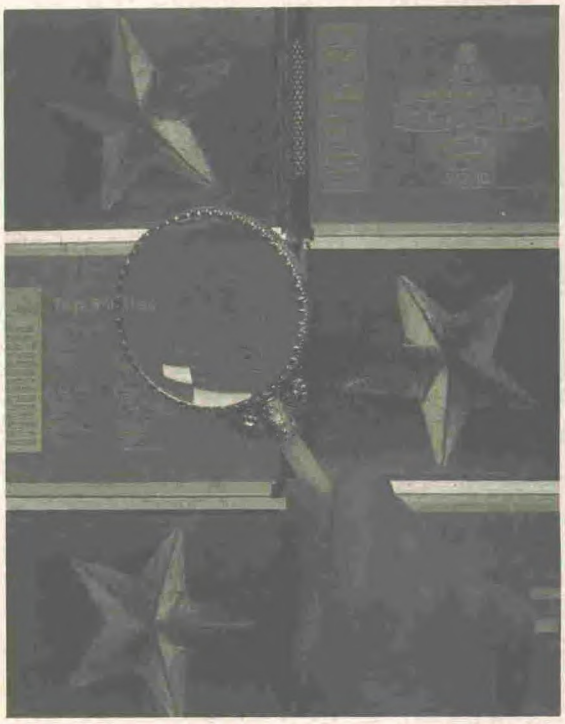
เมื่อกำหนดฟิลด์แต่ละตัวเรียบร้อยแล้ว ขั้นต่อไปก็คือใส่ฟิลด์เข้าไปในเทเบิลด้วยเมธอด Append ดังนี้

```
NewTbl.Fields.Append Fd1
NewTbl.Fields.Append Fd2
NewTbl.Fields.Append Fd3
NewTbl.Fields.Append Fd4
NewTbl.Fields.Append Fd5
NewTbl.Fields.Append Fd6
NewTbl.Fields.Append Fd7
```

การกำหนดอินเด็กซ์

การสร้างอินเด็กซ์ให้กับดาต้าเบสเป็นอีกสิ่งหนึ่งที่จำเป็น หากผู้อ่านเคยสร้างอินเด็กซ์ในวิชวลเบสิก 3 มาก่อนก็ขอให้ลืมวิธีการแบบเดิม ๆ ที่เคยใช้ไปได้เลยเพราะ ใน VB4 มีอะไรที่ดีกว่าเดิมให้ เมธอดที่จะใช้สร้างอินเด็กซ์ของ VB4 จะคล้ายกับการสร้างเทเบิลมากกว่าสร้างฟิลด์ คือ อินเด็กซ์จะต้องประกอบด้วยชื่อ, ฟิลด์ที่จะใช้เป็นอินเด็กซ์, กำหนดว่าจะใช้อินเด็กซ์ไหนเป็น primary index, กำหนดว่าฟิลด์ที่เป็นส่วนประกอบอันไหนยอมให้มีค่าซ้ำได้ ขั้นตอนการสร้างอินเด็กซ์มีดังนี้

1. ใช้เมธอด CreateIndex ของออบเจ็กต์ TableDef สร้างอินเด็กซ์ออบเจ็กต์



2. กำหนดค่าพรอพเพอร์ตี้ส์แต่ละอันของอินเด็กซ์ตามที่ต้องการ

3. ใช้เมธอด CreateField ของอินเด็กซ์ออบเจ็กต์เพื่อสร้างฟิลด์ออบเจ็กต์

4. กำหนดค่าพรอพเพอร์ตี้ส์แต่ละอันของฟิลด์ออบเจ็กต์ตามที่ต้องการ

5. ใส่ฟิลด์ออบเจ็กต์ที่กำหนดแต่ละอันไว้ในอินเด็กซ์ออบเจ็กต์

6. ใส่อินเด็กซ์ออบเจ็กต์ที่กำหนดแต่ละอันไว้ในออบเจ็กต์ TableDef

พรอพเพอร์ตี้ส์ของอินเด็กซ์ที่มักต้องกำหนดค่าก็คือ Primary ซึ่งถ้ากำหนดเป็น true ก็เท่ากับบังคับให้ค่าของฟิลด์นั้นมีค่าซ้ำกันไม่ได้ และเราจะใช้อินเด็กซ์นี้ในการค้นหาเรคอร์ดแต่ละอัน สำหรับพรอพเพอร์ตี้ส์ Unique นั้นมักใช้กับฟิลด์ที่ไม่ได้เป็น primary key โดยใช้บังคับไม่ให้ค่าของฟิลด์มีค่าซ้ำกัน นอกจากนี้ยังมีพรอพเพอร์ตี้ส์ของฟิลด์ออบเจ็กต์บางตัวที่เกี่ยวข้องกับการสร้างอินเด็กซ์ เช่น Attributes ที่ใช้กำหนดวิธีเรียงลำดับข้อมูล หากต้องการใช้วิธีจากมากไปน้อย (Z ไป A) ก็อาจกำหนดเป็น dbDescending หากต้องการสร้างอินเด็กซ์ที่เกิดจากการใช้ฟิลด์มากกว่าหนึ่งฟิลด์มาประกอบกัน เช่น ใช้ FirstName กับ LastName มารวมกันเป็นอินเด็กซ์ ก็ทำได้โดยใช้เมธอด CreateField ในวิชวลเบสิก 4 จะสร้างอินเด็กซ์ได้สูงสุด 32 ตัวต่อหนึ่งเทเบิล

```
Dim Idx1 As Index, Idx2 As Index, Fld1 As Field,
Fld2 As Field
Set Idx1 = NewTbl.CreateIndex("PersonID")
Idx1.Primary = True
Set Fld1 = Idx1.CreateField("PersonID")
Idx1.Fields.Append Fld1
Set Idx2 = NewTbl.CreateIndex("First_Name")
Idx2.Unique = False
Set Fld2 = Idx2.CreateField("FirstName")
Fld2.Attributes = dbDescending
Idx2.Fields.Append Fld2
NewTbl.Indexes.Append Idx1
NewTbl.Indexes.Append Idx2
```

โค้ดข้างบนนี้เป็นตัวอย่างในการสร้างอินเด็กซ์ออบเจ็กต์ในเทเบิล โดยเราเลือกฟิลด์ PersonID เป็น primary index ซึ่งต้องมีค่าไม่ซ้ำกัน ส่วนอินเด็กซ์อีกตัวที่กำหนดชื่อเป็น First_Name นั้น มีค่าซ้ำได้ เมื่อเรากำหนดส่วนต่าง ๆ ของเทเบิลเรียบร้อยแล้ว ขั้นตอนสุดท้ายก็คือต้องใส่เทเบิลที่สร้างเข้าไปในดาต้าเบสด้วยเมธอด Append ดังนี้

```
NewDb.TableDefs.Append NewTbl
NewDb.Close
```

```
Dim NewRel As Relation
Dim Fld1 As Field
'*************************
'Create the Relation object
'*************************
Set NewRel =
NewDb.CreateRelation("Person_Orders")
'*********************************
'Set the properties of the relation
'*********************************
NewRel.Table = "Persons"
NewRel.ForeignTable = "Orders"
'************************************************
'Create the relating field and set the
properties
'************************************************
Set Fld1 = NewRel.CreateField("PersonID")
Fld1.ForeignName = "PersonID"
'************************************************************
'Append the field to the relation and the
relation to the database
'************************************************************
NewRel.Fields.Append Fld1
NewDb.Relations.Append NewRel
```

การสร้างรีเลชัน

การออกแบบดาต้าเบสให้มีลักษณะรีเลชันนั้นมีความจำเป็นมาก เนื่องจากดาต้าเบสลักษณะดังกล่าวจะช่วยให้ดาต้าเบสที่สร้างเก็บได้อย่างมีประสิทธิภาพ ไม่มีการซ้ำซ้อนของข้อมูล สำหรับแจ็ทเอ็นจินการใช้งานดาต้าเบสแบบรีเลชันจะต้องสร้างรีเลชันออบเจ็กต์ขึ้นมาเก็บไว้ในดาต้าเบส รีเลชันออบเจ็กต์นี้จะบอกถึงความสัมพันธ์ระหว่างเทเบิลแต่ละอัน โดยใช้เทเบิลไหนเป็นหลักและติดต่อกันผ่านคีย์ฟิลด์ใด ขั้นตอนการสร้างรีเลชันระหว่างสองเทเบิลมีดังนี้

1. ใช้ประโยค Dim กำหนดตัวแปรรีเลชัน
2. ใช้เมธอด CreateRelation ของดาต้าเบสออบเจ็กต์สร้างรีเลชันออบเจ็กต์
3. กำหนดค่าพรอพเพอร์ตี้ส์ของเทเบิล primary และ foreign ของรีเลชัน
4. สร้างรีเลชันฟิลด์ของเทเบิลหลักด้วยเมธอด CreateField ของรีเลชันออบเจ็กต์
5. กำหนดพรอพเพอร์ตี้ส์ของฟิลด์ Foreign ของฟิลด์ออบเจ็กต์
6. ใส่ฟิลด์ที่กำหนดในรีเลชันออบเจ็กต์
7. ใส่รีเลชันออบเจ็กต์ไว้ในดาต้าเบส